

2006

Multimodal UAV ground control system

Thomas Batkiewicz
Iowa State University

K. C. Dohse
Iowa State University

Vijay Kalivarapu
Iowa State University, vkk2@iastate.edu

Thomas Dohse
Iowa State University

Bryan Walter
Iowa State University

See next page for additional authors

Follow this and additional works at: http://lib.dr.iastate.edu/me_conf



Part of the [Computer-Aided Engineering and Design Commons](#)

Recommended Citation

Batkiewicz, Thomas; Dohse, K. C.; Kalivarapu, Vijay; Dohse, Thomas; Walter, Bryan; Knutzon, Jared S.; Parkhurst, Derrick; Winer, Eliot H.; and Oliver, James, "Multimodal UAV ground control system" (2006). *Mechanical Engineering Conference Presentations, Papers, and Proceedings*. Paper 164.

http://lib.dr.iastate.edu/me_conf/164

This Conference Proceeding is brought to you for free and open access by the Mechanical Engineering at Digital Repository @ Iowa State University. It has been accepted for inclusion in Mechanical Engineering Conference Presentations, Papers, and Proceedings by an authorized administrator of Digital Repository @ Iowa State University. For more information, please contact digirep@iastate.edu.

Authors

Thomas Batkiewicz, K. C. Dohse, Vijay Kalivarapu, Thomas Dohse, Bryan Walter, Jared S. Knutzon, Derrick Parkhurst, Eliot H. Winer, and James Oliver

Multimodal UAV Ground Control System

Thomas J. Batkiewicz¹, K.C. Dohse¹, Vijay Kalivarapu¹, Thomas Dohse¹, Bryan Walter¹, Jared Knutson¹,
Derrick Parkhurst², Eliot Winer³, and James Oliver⁴
Iowa State University, Ames, IA, 50011, USA

As unmanned units become more capable of self-control, and as their integration into our military forces increase, the role of the operator of these units is going to change. Current UAV ground control systems typically require too much of the operator's attention per unit, and the role of the future operator will be more like that of a manager than that of a pilot. This paper describes research being done on a user interface for a future ground control system. Beginning with a visualization of a virtual battlefield, this next-generation UAV ground control system incorporates all available information and analyses in combination with a synthetic battlespace in a context-relevant manner. This marriage of synthetic and real information feeds, within a single visualization space, is designed to increase the situational awareness of a single operator. This virtual battlefield provides the operator with the information necessary to accomplish their task, with multiple modes of interaction for the user. The UAV Ground Control System detailed here utilizes a speech command interface, a wireless joystick interface, as well as a tablet-based direct manipulation interface similar to those used in general air unit ground control systems. Through these different types of interactions, an operator is presented with various sources of data to manage and command military units as clearly and simply as possible.

I. Introduction

Currently, UAVs specialize in missions commonly labeled "the dull, the dirty, and the dangerous", but the DOD Roadmap projects that future UAVs will be capable of performing a wide range of combat missions¹. The ever-increasing complexity of the UAV, as well as the dynamic role it will play in military operations, provides a serious design challenge when considering its control interface. The ability for a single operator to control and monitor several UAVs is chiefly dependent on the resources of the operator's attention. To fully harness the capabilities of UAVs, the role of the operator must change from that of a pilot to that of a manager, with a single operator in charge of multiple UAVs¹.

This drastic change in the responsibilities of the operator has some serious implications. First, the older "soda straw" methods for UAV control will not be adequate for the management of multiple UAVs. This involves controlling a UAV remotely using 2D images captured from a camera mounted on the vehicle. A user only has a limited view and must judge perspective and distance through purely visual cues. In this context, constant micromanagement of individual units will quickly overwhelm an operator's attentional capacity. A wider perspective will be required for the operator, providing them with information on all the UAVs in their engagement theater, as well as other information from joint forces, intelligence, and other sources². Furthermore, the ability to direct UAV swarms; providing high-level commands to large numbers of UAVs will be of use when controlling the UAVs of the future³. Operators need a high-level portal to the engagement as a whole to provide them with the situational awareness to properly utilize the UAVs they control².

¹ Research Assistant, Human Computer Interaction, Virtual Reality Applications Center, 1620 Howe Hall, Iowa State University, Ames, IA, 50011, USA.

² Assistant Professor, Department of Psychology, Human Computer Interaction, Virtual Reality Applications Center, 1620 Howe Hall, Iowa State University, Ames, IA, 50011, USA.

³ Assistant Professor, Department of Mechanical Engineering, Human Computer Interaction, Virtual Reality Applications Center, 1620 Howe Hall, Iowa State University, Ames, IA, 50011, USA, member.

⁴ Professor, Department of Mechanical Engineering, Human Computer Interaction, Virtual Reality Applications Center, 1620 Howe Hall, Iowa State University, Ames, IA, 50011, USA, member.

However, maintaining the ability for an operator to drill down to a single UAV to get detailed information and take near-complete control will remain valuable in protecting unmanned assets⁴. It is likely that human decision making will always be a part of UAV control, especially as UAVs move beyond reconnaissance and surveillance and into armed combat. The decision to fire/attack will still come from a human source. Thus, the presentation of low-level details is crucial, but must not overwhelm the operator. Managing these issues is inherently a multidisciplinary analysis and decision-making process with a human-in-the-loop. The data supplied to a user ranges from geographic to engineering analysis and optimization results to discrete event simulation. A user interface for this type of command and control must fuse, index, and search all these sources so pertinent information is readily available for real-time decision-making. Overall, there are several key components to a next-generation UAV ground control system:

- 1) Many-to-one relationship between UAVs and operator.
- 2) Variable contexts for information flow to the operator; from comprehensive to specific.
- 3) Multiple control paradigms enabling swarm-wide and individually-detailed interaction methods.
- 4) System capability to direct the operator's attention where needed.

Considering these requirements, this paper describes the development of a next-generation UAV ground control system. Relying primarily upon a synthetic view of the world, and augmented by real sensor information coming from military units in theater, this system provides data and representations on an as-needed basis, providing ample but not overburdening amounts of information to the operator.

II. UAV Ground Control System

The software interface described in this paper has been termed the Virtual Battlespace. Battlespace is a 3D visualization application that provides dynamic viewing and interaction with units distributed over a large battlefield. Battlespace is built upon the premise of using a primarily synthetic environment to build a baseline 3D visualization and integrating live sensor feeds directly into this environment, constantly updating and inserting the most up-to-date information into a fused display. By placing perspective-sensitive data, such as a camera feed in context, the cognitive load on the operator is decreased. The Battlespace application can be run on a generic desktop display, but truly works best on highly immersive CAVE displays; the larger field of view provided by an immersive environment reinforces the desired increase in an operator's context-awareness.

The goal of this approach is to maximize the user's situational awareness by combining what would typically be several applications into one. However, since the Battlespace visualization is designed towards improving an operator's situational awareness, it incorporates many features that move a user away from a traditional direct manipulation interaction scheme. However, many of these features such as navigation, selection, and manipulation in 3D of specific entities is not a trivial task. Thus, a second interface was constructed to complement the Virtual Battlespace, the Battlespace Task Manager.

The Task Manager is designed to be used alongside Battlespace, augmenting its viewing and interaction by providing a direct manipulation interface to the units in the engagement. The Task Manager provides a birds-eye north-up map display, that can be utilized either on a desktop or tablet PC. Based on the map displays currently used by UAV operators, the Task Manager provides a mission overview as well as the ability to select, review, and modify waypoints, time-of-arrivals, and other unit parameters. Additionally, the Task Manager can be used on its own, as a mission planning tool, in a fashion similar to Falconview, an application utilized by some branches of the armed forces⁵. The following sections of the paper detail the Battlespace and Task Manager application along with the various input devices used in the environment.

III. Battlespace Visualization

The Battlespace application operates in two primary modes; a strategic mode providing a "god's-eye view" with symbolic representations of units, and a pilot mode that enables visualization of units in a realistic scale and provides a chase-cam view. Changing between the two is a seamless, context-preserving, view-interpolating transition that allows the user to maintain a high level of situational awareness. All of the various interaction devices can be used in both modes to adjust the visualization and direct the units.

1. Strategic Mode

In strategic mode, the application displays the battlefield from a god's-eye view, providing a high-level view of unit's positions and goal states. Typically, the viewpoint is set several miles above the earth's surface to provide full coverage of the engagement. The operator can manipulate the viewpoint to any location and orientation, allowing them to choose what to see. This presentation is aimed to maximize the situational awareness of the user, but since

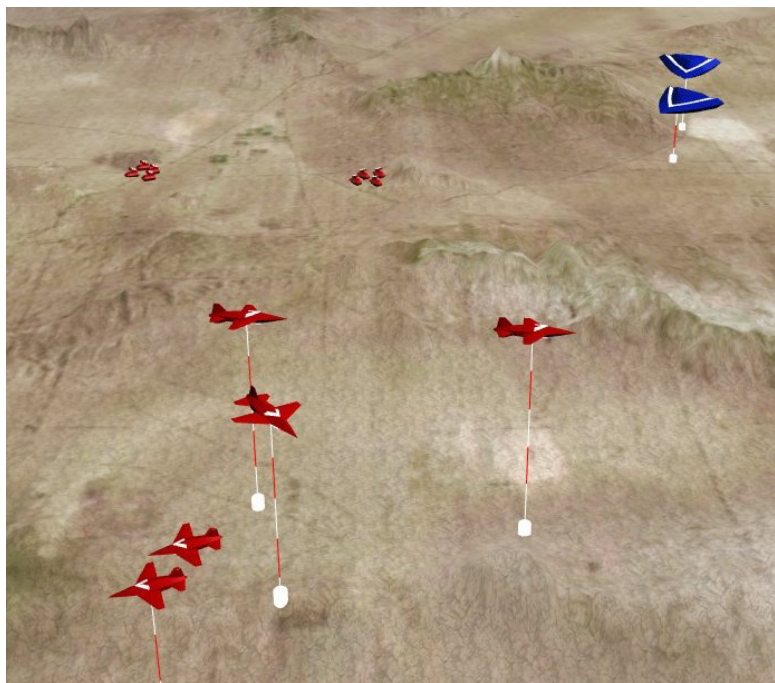


Figure 1. Strategic mode with height sticks.

it relies on a 3D visualization there are some key factors that must be taken into account so that the information is not ambiguous.

One of the primary ways in which mission commanders view engagements is from 2D, top-down displays that overlay altitude information onto them. However, from an arbitrary viewpoint in a 3D visualization, this information may be misleading or deceptive, causing the operator to misunderstand the situation. Since the ground position and altitude can be difficult to interpret with the visualization of a vehicle in midair, we developed the height stick to compensate for this. With alternating bands of red and white, this stick extends from the bottom of the unit down to the ground. Each band represents 5000 feet, and so a rough measure of altitude can quickly be determined at a glance. Further, to identify ground position, a small cylinder was placed at the base of the height stick to help the operator identify the ground position directly below the air vehicle. A view of several units in strategic mode, with height sticks enabled is shown in Figure 1.

Another visual feature that helps the operator maintain situational awareness is the history trails. History trails are used to show the path each vehicle or set of vehicles has followed. A colored path is overlaid directly onto the terrain rendering which denotes the past positions of the entities. The colors are used only to indicate whether the trail leads to friendly (blue) or hostile (red) forces. This allows an operator who has spent time examining one section of an engagement in detail to quickly catch-up to the current situation.

For unmanned entities, the future flight path may also be displayed in the Battlespace. Since the altitude of the waypoints along the path matter, this must be done in 3D. Similar to the method for height-sticks, the future flight paths are shown with alternating bands each corresponding to 5000 feet of elevation. Waypoints are drawn as vertical columns along the path, creating a visual structure that resembles a fence. This is shown in Figure 2.

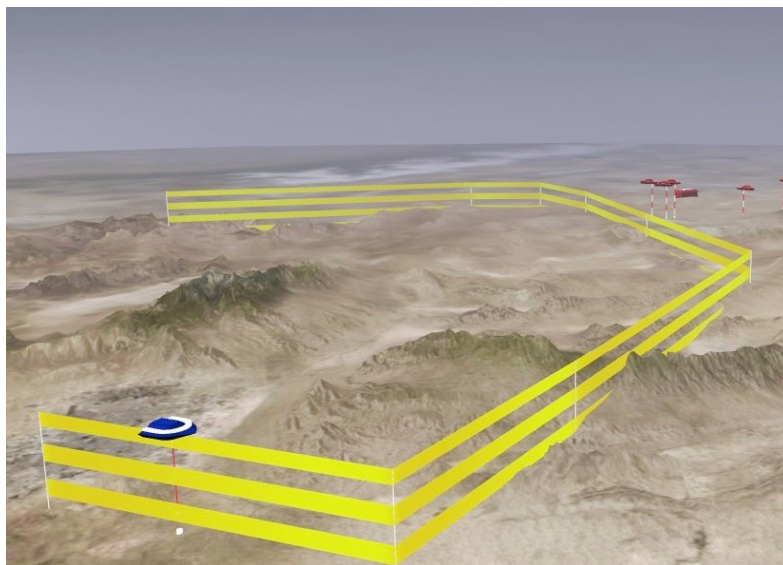


Figure 2. Strategic mode with UAVs and height bands.

Users can also utilize a heads-up display (HUD) that provides a north-up 2D radar view. It shows current unit positions and orientations in a birds-eye view, and uses a fixed-position reference for the user's position, with a semi-transparent white overlay to indicate the primary viewing frustum (e.g., the front wall of an immersive CAVE display). All this information is layered upon a map that provides terrain/political data for the immediate area. This is shown in Figure 3.

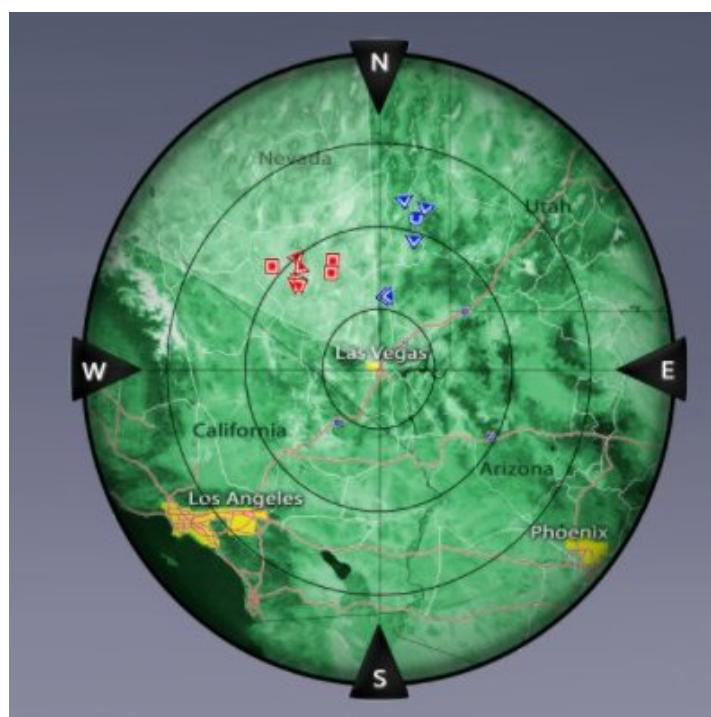


Figure 3. HUD radar

Another key feature of the Battlespace application is its approach to displaying the properties and attributes of units in the engagement to the operator. The operator's knowledge of the specific capabilities of their units, as well as those of hostile forces, is important to maximize combat effectiveness. Visualizing unit-specific information rather than information regarding the unit's position, plan, or the general flow of battle is important to providing the necessary details required for command and control.

The ability for an unmanned unit to act independently of an operator is strongly based in the capabilities of its sensors. Sensors onboard unmanned vehicles, as well as weapons systems may have maximum ranges to which they are effective. The ability for an operator to understand what the UAV is able to "see" will be an important factor in command and control, especially as the level of autonomy grows. Furthermore, the display of weapons capabilities for both friendly and hostile units can be helpful for an operator. In Figure 4, this display of threat and sensor information is shown: the hemispheres of overlapping concentric red rings indicate the airspace threatened by a group of surface-to-air missile launchers, and the green wedge indicates a forward-looking sensor on the lead incoming unit.

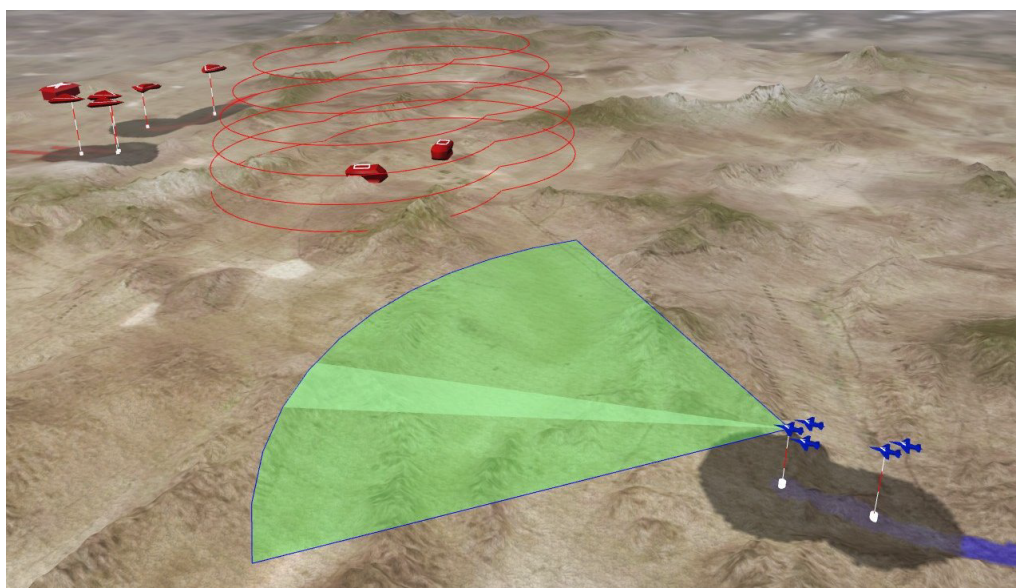


Figure 4. Sensor visualization in strategic mode.

Another primary source of information from an unmanned vehicle is the cameras mounted on it. Relaying the picture back to the operator is standard practice in current pilot-oriented interfaces, and should remain integrated in future interfaces. One of the ways this information is handled in Battlespace is to register sensor imagery to a location and provide a specific symbol for it. If the image sent back by the UAV is noteworthy, perhaps designated as a potential threat by automatic target recognition (ATR) or from some other source, the operator will need to know this. Using the alert system previously mentioned an operator's attention can be called to view this new development. If the ATR was unable to make a determination, the operator can examine the video information and classify it.

This approach is suitable for situations in which the UAV has already flown past this location and gone on to other tasks, but real-time streaming camera information is handled differently. Placing the camera feed in context in this 3D environment would be an extension of streaming video's use in current unmanned ground control systems, augmenting the soda-straw view with our synthetic terrain. However, this information is specific to the viewpoint of the unit, and will be handled in the other main mode of Battlespace, the pilot mode.

2. Pilot Mode

Pilot mode is the second operational mode for the Battlespace application. In this mode, vehicles are rendered at a realistic scale, and generally the operator's viewpoint is localized to a specific unit, with the default being a chase-plane view of the currently selected unit. Other fixed viewpoints are provided that give a variable distance to the unit, as well as providing left and right-looking views. Since the vehicles in pilot mode are rendered at a realistic scale, vehicles not close to the viewpoint are difficult to see due to distance, as in real life. A colored (red/blue)

hemisphere is used to indicate position of such remote units, and the height sticks described in the strategic mode section are also used. These features are shown in Figure 5.

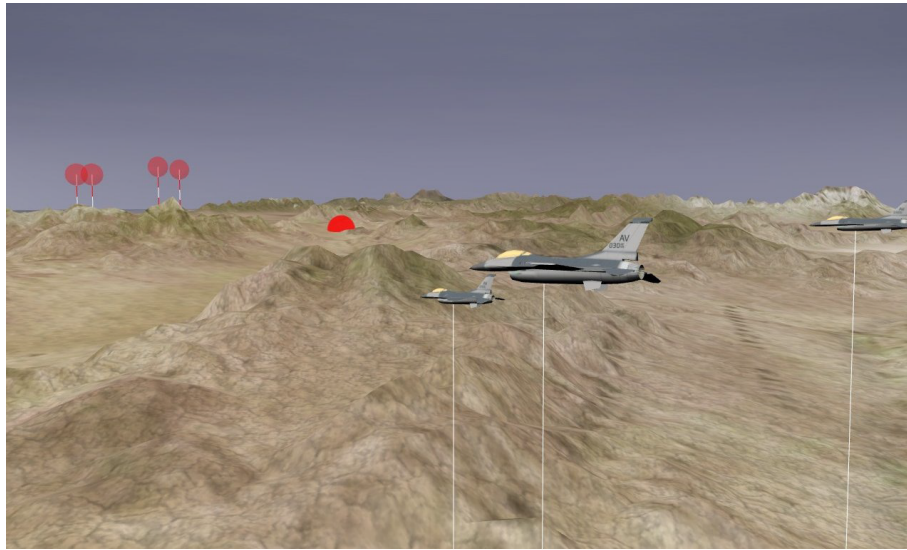


Figure 5. Pilot Mode in Battlespace

Also in pilot mode, precise values can be provided for specific vehicle information such as altitude, speed, heading, etc. All the things that would typically be included in a pilot's HUD can be displayed in pilot mode. Further, camera information can be added in context, so that its orientation relative to the vehicle can be replicated in the simulated environment, overlaying the synthetic information with the most current information available.

Another capability that pilot mode deals with is the issue of 3D path generation, visualization, and planning. When a unit is approaching a threat, the operator may wish to choose a new path that minimizes exposure, maximizes speed, or follows some criteria to achieve an objective. An algorithm can be run to generate alternate paths, but ideally the operator should be allowed to decide which path best achieves the objectives. Upon selection the path is updated and the unit will follow its new path⁶. Multiple paths can be generated with a user selecting which one the vehicle will follow. This is shown in Figure 6.

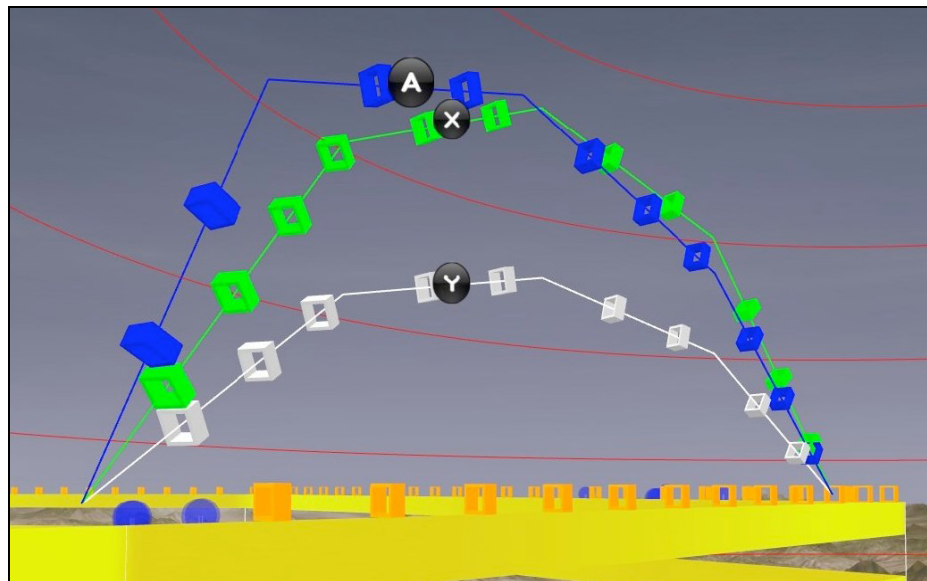


Figure 6. View of three resulting alternate paths generated to avoid a single threat.

Lastly, pilot mode provides the displays necessary for teleoperation. With an egocentric viewpoint with respect to the unit, a pilot can have all the sensors and instrumentation displays integrated into this synthetic environment. Rather than having various instrumentation displays scattered in front of them, a single fused display can provide them the data required to fly the aircraft remotely. Basing it on the virtual environment allows for the commander to take control, whether directly or indirectly, of the unmanned vehicle as the situation demands.

3. Alert System

In general, the operator is unrestrained in their ability to view the battlefield, allowing the operator to decide what areas should receive their attention. However, there are times when emergency situations arise that require an operator's immediate attention. The chief way that the operator's attention is called to areas requiring their input is through the alert system present in Battlespace. The alert system is a graphical notification system that can be triggered by a number of different issues such as low fuel, threat/target proximity, etc.... Any time an entity enters into a situation that requires an operator's attention, a visual notice appears on the display. One button press will take the operator to the new alert and present them with the details of the problem, as well as the information required to make a decision. An example of an alert is shown in Figure 7.



Figure 7. A threat alert display

IV. Task Manager

The Task Manager provides a 2D direct manipulation map interface to the units in the engagement. Using a birds-eye perspective, icons are drawn at the ground position of the unit (or its waypoints) and altitude is indicated with a text display. When considering the overall layout of the application, special attention was paid to existing mapping applications and 2D interface design principles. The largest portion of the screen was devoted to the map, since the north-up display is the primary source of information for the user, as well as the direct-manipulation interface to the units. Below the map display is a timeline display, that can be scrolled horizontally to view the planned times for units to arrive at waypoints as well as time-on-target information. The top and right side were used for the placement of application controls, with pull-down menus and a toolbar across the top. At the very bottom of the application is a status bar that is used to provide some information to the user as to any background processes the application may be carrying out, as well as providing confirmation of unit information on selection/addition. A screenshot of the Task Manager is shown in Figure 8.

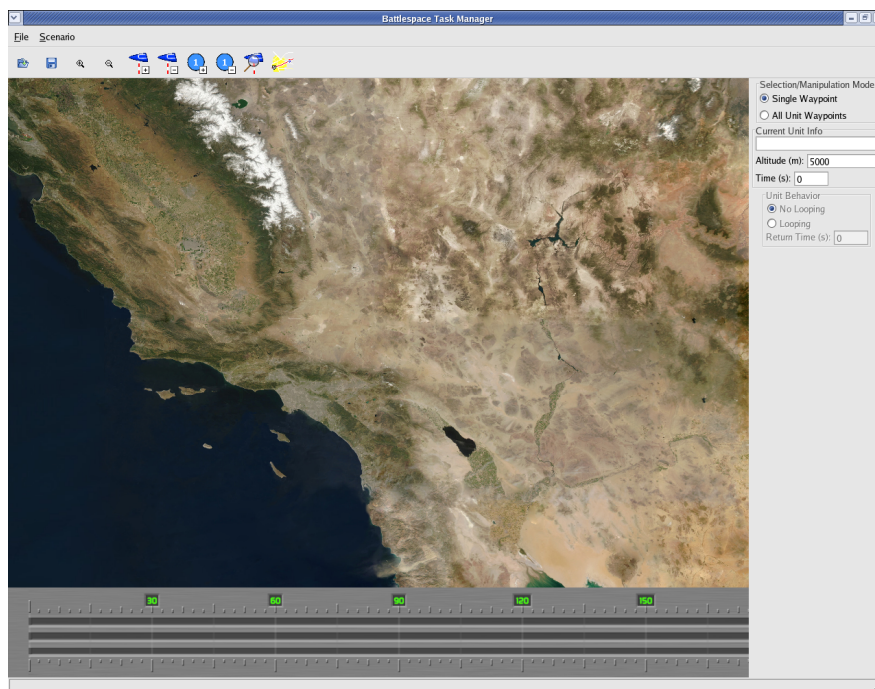


Figure 8. Task Manager interface

In the figure, a map display shows a birds-eye, north-up view of a single-resolution satellite image of the southwest United States. Any satellite image or graphic can be input and displayed in the Task Manager. The eye-height above sea level can be adjusted by the user, allowing them to zoom in and out and allow for more precise interactions. Other images containing political boundaries, cities, and some information detailing roads can be overlaid on top of the satellite imagery. The positions of a unit's waypoints can be shown on top of the terrain, and are connected by a line to indicate their path. The waypoint information for multiple units is displayed all at once on a single map. The path segments between waypoints will not simply be represented by a single-pixel line; a polygon, textured to match the unit's waypoints will be used. The reasoning behind this is that since waypoints have time-of-arrival constraints, the segments between must have speed constraints. The goal is to make these segments large enough to be selectable, such that an operator can input speed, and to add sensor-positioning controls to the segments.

Additionally, unit positions from a radar track or GPS can be displayed on top of this path information to indicate progress towards mission goals. It is important to note that one element that has not been included thus far is the track or link quality to that unit. In real engagements, the quality of the information coming from the unit or from the radar source is an important piece of information for the operator, but it would require additional work beyond the scope of the interface to simulate this behavior. As such, this interface assumes a high-fidelity connection between the operator's interface and the incoming signal feeds. Displaying this information is a relevant interface challenge, but since the existing simulations used in the development process do not carry this information, time was not spent specifically examining these elements.

Below the map display is a timeline feed that shows projected waypoint arrival times for the currently selected unit. Modifications of unit waypoints in the map display are automatically taken into account; the effect that changes in distance between waypoints, adding waypoints, and removing waypoints have on the arrival times are all recalculated on the fly and displayed in the timeline display. Time on target information is also shown on the timeline display and is indicated by a red box. Multiple 'tracks' appear in the timeline display, allowing multiple units to be selected to coordinate arrival and firing times on either the same target or different ones.

The operator can accomplish a variety of tasks in multiple ways; by using buttons on the toolbar or control panel, or by clicking points and icons in the map and timeline, operators can select a unit, modify its information, adjust its path, and update timing information. Supporting multiple interaction methods was important since this application was designed to be used on two very different systems – desktop PCs with keyboard/mouse interaction, and tablet PCs with pen-based interaction.

V. Interaction Devices

Input to the Battlespace application is handled in a number of different ways. Though keyboard/mouse interactions would be appropriate in a desktop setting, they require a surface to operate upon which would limit the user's field of view in an immersive environment. Defining a standard set of input devices will help us measure the effectiveness of display. With that in mind, we set out to find some interaction devices that would utilize the input capacity of both hands simultaneously but do not have a large physical footprint, and would allow for rapid input for time-critical decision making. Not surprisingly, our first input device is borrowed from another real-time decision making domain; video games.

A. Gamepad Controller

The gamepad input device is a wireless Logitech game controller that has two analog joysticks, an analog sliding throttle control, a digital direction-pad, seven usable primary buttons and four shoulder buttons. Two of the buttons on the controller are unusable by the application since they are used to toggle hardware features on the controller only. An image of the controller is shown in Figure 9.

One of the key features of the gamepad interface is that although there are only 11 usable buttons, there are a far greater number of interactions possible based on the implementation of the interface. Internally, the application maintains a keyed table of button signals to function pointers. These function pointers can be reassigned upon receiving commands from within the Battlespace and also by the functions called when a button is pressed. For example, this means that the 'A' button can be used to trigger selection mode, but upon selection, the 'A' button can then be used within this selection mode. Buttons can be remapped on the fly to achieve desired interactions. This feature keeps the buttons closest to the resting position of the user's hand available for constant input which makes for quicker interactions. This also allows the buttons at the fringe of the device to be saved for specific functions that don't change over time, such as responding to alerts (which may decrease reaction time since it is always the same button).



Figure 9. Gamepad Controller

Though the controller interface has these benefits, in itself the controller provides no visual feedback to the user. Requiring the user to memorize all the possible use modes is not a viable strategy, so other components must be integrated to provide a meaningful interaction scheme. A menu system provides a visual listing of actions the user could take and is a fairly well-developed scheme that finds use in everyday applications.

B. Menu System

Menu systems are very common and simple interaction methods, since menus provide a visual reference and confirmation of your actions. In Battlespace, the menu is rendered in 3D as a billboard, so that it always faces the user. From the menu, users can toggle visualization features on and off, pause or slow a simulation replay, and view/edit targeting information using the controller. These various categories are grouped into separate panels in the menu. Input to the menu is done through the gamepad controller, and each menu provides a visual key to for use of the buttons that are related to that menu. The shoulder buttons on the controller are used to switch between menu pages, and the directional pad is used for selection within a single page.

C. Voice

Another input device that is easily integrated via wireless networking is the input of voice commands. A wireless headset microphone can be worn by the user and, using a speech-to-text engine, the user's spoken words can be parsed and translated into commands. One of the most useful features of the voice input system is the ability for users to 'tag' a unit with a particular designation. Units can be given specific names by the user that can then be understood by the system when combined with existing commands. For example, a unit designated 'alpha', could then be selected when the user says 'select unit alpha'. This type of interaction reinforces the idea that Battlespace provides a level of control greater than that of a typical pilot and more on the level of a commander.

D. Tablet PC

When using the Battlespace application on a desktop display, pairing another machine to run the Task Manager on an adjacent monitor or LCD screen is a logical solution. However, when the Battlespace is visualized in an immersive environment that utilizes a display with a large field of view, obscuring that view is counter-productive.

The use of a wireless, hand-held tablet PC allows the user to maintain a 2D interaction scheme, using a touch-sensitive pen input to interact directly with the screen of the tablet, without interfering with their view of the Battlespace. Specifically, the Toshiba Tecra M4 was used in development of this interface. The Task Manager was designed specifically to handle this shift in modality; desktop to tablet, however other factors make this situation less than ideal. The weight of a tablet PC (6.2 lbs. in the case of the Tecra M4) is nowhere near that of a desktop system, however it is not negligible and will quickly fatigue the user and prohibits long-term use.

E. Touch Table

One solution to this problem is to replace the tablet with a table-sized display device that sits in front of a multiple-walled projection system. Putting the Task Manager on this type of device will reduce the strain on the user's arms, as well as provides a single location for the operator to sit at. Allowing for the user to directly touch the screen, either with a stylus or their fingers mirrors the interaction already existing in the tablet PC interface. Additionally, the size of the display can be noticeably increased. Based on the work of Jeff Han, a touch-sensitive table display was built recently, and the integration of this device into the overall system has just begun⁷.

VI. Conclusion

Overall, this interface represents a fundamental shift in the relationship between the operator and UAV, and provides an experimental tool in gauging what types of technology will need to be developed to support the operator in their new role. Developing the next generation of command and control software interfaces to deal with the advancement in UAV autonomy will be crucial in the deployment of unmanned systems in the future. This work is ongoing, and is not intended as an ultimate solution for the operator interface, but rather is a solid first step in investigating potential interfaces for the complex task of managing unmanned vehicles.

Currently, there are user studies being conducted using this software interface as the research platform. One ongoing study is delving into the situational awareness provided by an immersive environment. The number of screens, their resolutions, and the size in the user's field of view are all critical factors in determining how much context is necessary without overwhelming the user. Another potential study currently being considered is a competitive gaming experiment utilizing the research platform; modifying the display and interaction devices to create a number of different conditions, and then having subjects control opposing forces to determine effectiveness.

Acknowledgments

The authors would like to thank the Air Force Office of Scientific Research (AFOSR) for their financial support of this project. Further, the following open source software packages were used in the development of this interface and we would like to give them all credit.

- 1) VRJuggler Virtual Reality Toolkit⁸
- 2) OpenSG Open Source Portable Scenegraph⁹
- 3) Demeter Terrain Engine¹⁰
- 4) wxWidgets GUI API¹¹

References

- ¹US Department of Defense (2002). "Unmanned Aerial Vehicles Roadmap 2005-2030." URL: <http://www.acq.osd.mil/usd/Roadmap/Final2.pdf> [cited 14 July 2006].
- ²Knutzon, J., Walter, B., Sannier, A., & Oliver, J. "Command and Control in Distributed Mission Training - An Immersive Approach." *NATO Symposium on Critical Design Issues for the Human-Machine Interface*, Prague, Czech Republic, May 2003.
- ³Walter, B., Sannier, A., Reiners, D., & Oliver, J. "UAV Swarm Control: Calculating Digital Pheromone Fields with the GPU." *IITSEC*, Orlando, Florida, December 2005.
- ⁴Walter, B., Knutzon J., Sannier A., Oliver J., VR Aided Control of UAVs. 3rd AIAA Unmanned Unlimited Technical Conference, Paper AIAA 2004-6320, Chicago, 2004.
- ⁵FalconView Mapping System, Software Package URL: <http://www.falconview.org/> [cited 14 July 2006]
- ⁶Knutzon, J. "Managing Multiple UAV's from a 3D Virtual Environment." Ph.D. dissertation, Human-Computer Interaction, Iowa State University, Ames, IA, 2006.
- ⁷Han, J. Y. "Low-Cost Multi-Touch Sensing through Frustrated Total Internal Reflection." *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*, 2005.
- ⁸VRJuggler Virtual Reality Toolkit, Software Package, Version 2.0.1 URL: <http://www.vrjuggler.org/> [cited Jan 2006]
- ⁹OpenSG Open Source Portable Scenegraph, Software Package, Version 1.4 URL: <http://www.opensg.org/> [cited May 2006]
- ¹⁰Demeter Terrain Engine, Software Package, Version 3.21 URL: <http://www.tbgssoftware.com/> - [cited Jan 2006]
- ¹¹wxWidgets GUI API, Version 2.6.2 - <http://www.wxwidgets.org/> - [cited April 2006]